



This is the author's final draft post-refereeing (post-print)

Find more peer-reviewed articles on our open access repository:
<http://hal-univ-rennes1.archives-ouvertes.fr/>

Full 4-D quaternion discrete Fourier transform based watermarking for color images

Beijing Chen^{a,b,c}, Gouenou Coatrieux^c, Gang Chen^d, Xingming Sun^{a,b},
Jean Louis Coatrieux^{e,f,g}, Huazhong Shu^{d,e,*}

^a School of Computer & Software, Nanjing University of Information Science & Technology, Nanjing 210044, China

^b Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science & Technology, Nanjing 210044, China

^c Institut MINES-TELECOM, TELECOM Bretagne, Inserm UMR1101 Latim, Brest 29238, France

^d Laboratory of Image Science and Technology, Southeast University, Nanjing 210096, China

^e Centre de Recherche en Information Biomédicale Sino-Français (CRIBs), Nanjing 210096, China

^f INSERM, U1099, Rennes, F-35000, France

^g Laboratoire Traitement du Signal et de l'Image, Université de Rennes I, Rennes 35000, France

ABSTRACT

Among the few existing color watermarking schemes, some use quaternion discrete Fourier transform (QDFT). By modulating at least one component of QDFT coefficients, they spread the watermark over two or three of the RGB color channels. However, these schemes do not fully utilize the four-dimensional (4-D) QDFT frequency domain and some also suffer from a watermark energy loss directly at the embedding stage. In this paper, we first establish the links that exist between the DFT of the three RGB color channels and the components of QDFT coefficients while considering a general unit pure quaternion. Then, for different unit pure quaternions i , j , k or their linear combinations, we discuss the symmetry constraints one should follow when modifying QDFT coefficients in order to overcome the previous drawbacks. We also provide a general watermarking framework to illustrate the overall performance gain in terms of imperceptibility, capacity and robustness we can achieve compared to other QDFT based algorithms, i.e. when fully considering the 4-D QDFT domain. From this framework we derive three schemes, depending on whether i , j or k is used. Provided theoretical analysis and experimental results show that these algorithms offer better performance in terms of capacity and robustness to most common attacks, including JPEG compression, noise, cropping and filtering and so on, than other QDFT based algorithms for the same watermarked image quality.

1. Introduction

With the rapid development of communication and information technology, security has attracted much attention. For multimedia documents like images or videos, watermarking is a promising solution for broadcast monitoring, access control, copyright protection and so on. The key idea behind digital watermarking is to leave access to the document while maintaining it protected by means of an imperceptible watermark inserted or added to it. Watermark insertion is based on the principle of controlled perturbation of the host document. If the watermark should be unobtrusive, it can also be made robust to host document modifications [1].

Although many algorithms have been proposed to deal with grayscale images, only a few of them address color images. Nevertheless, by spreading the watermark over the three RGB (Red Green Blue) color channels rather than simply considering the image luminance, one may expect two advantages [2,3]: the potential amount of embedded data can be greater; the watermark robustness can be increased while better preserving the fidelity/quality of the image.

Quaternions, which have been more and more used in color image processing in the past two decades, offer an interesting solution to achieve this goal. They represent an image by encoding its three color channels on the imaginary parts of quaternion numbers [4–16]. The main advantage of such a representation is that a color image can be processed holistically as a vector field [5–7]. So does color image watermarking [8–16], which mainly makes use of the quaternion discrete Fourier transform (QDFT). By modulating at

* Corresponding author at: Laboratory of Image Science and Technology, Southeast University, Nanjing 210096, China. Fax: +86 25 83 79 26 98.

E-mail address: shu.list@seu.edu.cn (H. Shu).

least one component of QDFT coefficients, the watermark is spread throughout two or three of the RGB channels [9].

QDFT watermarking was first introduced by Bas et al. in [8], where they compare the fidelity of color watermarked image considering two unit pure quaternions $\mu_{perc} = (-2\mathbf{j} + 8\mathbf{k})/\sqrt{68}$ or $\mu_{lum} = (\mathbf{i} + \mathbf{j} + \mathbf{k})/\sqrt{3}$ in the case of message bit stream embedding by means of the quantization index modulation (QIM) [17]. It should be noticed that μ_{lum} is the most common unit quaternion used in the quaternion based image processing literature. Since the preliminary work of Bas et al., Tsui et al. [10,11] propose a non-blind algorithm (i.e. the original host image is required for watermark extraction) that inserts a sequence of bits into the QDFT domain derived from the $L * a * b$ components of the image. In fact, they modulate a color watermark pattern added to blocks of QDFT coefficients. On their side, Ma et al. [12] propose a semi-blind color watermarking scheme based on local QDFT. They first utilize an invariant feature transform so as to get some significant points or features of interest, and embed a binary watermark into QDFT coefficients of blocks centered on each of these points. Sun et al. [14] suggest embedding a binary image into the amplitude of QDFT AC coefficients of 8×8 pixel blocks. Later, they combine QDFT and quaternion singular value decomposition into a semi-blind color image watermarking (original singular values of the host image are needed for watermark extraction) [15].

Recently, Jiang et al. [9] point out that, in [8], Bas et al. do not consider the precondition that all the real parts of the quaternion matrix which represents the image in the spatial domain are required to be equal to zero. Thus, applying the inverse QDFT (IQDFT) to QDFT coefficients that have been watermarked without care may lead to a quaternion matrix with non-null real parts. Then, only taking the three imaginary parts of this quaternion matrix so as to get the watermarked image induces a watermark energy loss. More clearly, even without attacking the image, the extracted watermark differs from the one embedded in the QDFT domain. Watermark detection is compromised. In the case of μ_{lum} , Jiang et al. show that this issue can be solved by modifying QDFT coefficients symmetrically. Under this constraint, they insert a binary watermark by applying QIM to QDFT coefficients [9]. Notice that [10–12] also satisfy this constraint of symmetric distortion. The scheme of Jiang et al. has been extended into a non-blind watermarking scheme by Wang et al. [13] with the embedding of a grayscale image watermark. Wang et al. further improve this scheme making it robust to geometrical distortions with the help of least squares support vector machine and pseudo-Zernike moments [16].

Experimental results given in these papers show that quaternion-based algorithms can achieve a reasonable trade-off between fidelity and robustness. However, as we will see in Section 3, they do not fully take advantage of the four-dimensions (4-D) of the QDFT domain. Indeed, these previous works only consider from one to two components of QDFT coefficients but not all of them. Even though these algorithms can be used to cover independently all of the four components of QDFT coefficients, the above mentioned precondition as well as the image distortion may be more difficult to handle.

In this work, considering the case of different unit pure quaternions \mathbf{i} , \mathbf{j} , \mathbf{k} or linear combinations of them, we demonstrate the conditions one must satisfy when modulating QDFT coefficients so as to: (1) avoid watermark energy loss at the embedding stage; and, (2) take full advantage of the 4-D QDFT domain in terms of watermark capacity and robustness.

The remaining of this paper is organized as follows. Section 2 comes back on the quaternion representation of color images, and on the definition of QDFT. In Section 3, considering the general definition of a unit pure quaternion (i.e. $\mu = \alpha\mathbf{i} + \beta\mathbf{j} + \gamma\mathbf{k}$, $\alpha, \beta, \gamma \in \mathbb{R}$, $\|\mu\| = 1$), we explain how to avoid the watermark energy loss

problem, i.e. the symmetry update constraints of QDFT coefficients, through establishing the relationships that exist between the different QDFT coefficient components and the conventional discrete Fourier transform (DFT) of the RGB color channels. We then analyze watermarking capacities one can expect using the different unit pure quaternions \mathbf{i} , \mathbf{j} or \mathbf{k} compared with others (i.e. μ_{perc} and μ_{lum}). After these preliminaries, Section 4 details the experimental framework we consider for evaluating the gain of performance in terms of watermark imperceptibility and robustness when the four dimensions of the QDFT domain are holistically used. This framework corresponds to a blind color watermarking system within which the modulations of Bas et al. [8] and of Jiang et al. [9] as well as of Wang et al. [16] have been included for fair comparison. It is important to notice that, in this work, we exploit an improved version of the solution proposed by Bas et al. to make it respect the above mentioned precondition. Indeed, this latter can be satisfied for μ_{perc} if coefficients are modified accordingly. Experimental results are given and discussed in Section 5, where we further compare our approach with the method reported in [3], a more conventional, recent and efficient solution based on Schur's decomposition and which treats RGB channels independently. Section 6 concludes the paper.

2. Some preliminaries

In this section, we come back on the definition of quaternion before introducing the QDFT.

2.1. Quaternion number and quaternion representation of color images

Quaternions were introduced by the mathematician Hamilton in 1843 [18]. They are generalizations of complex numbers. A quaternion number can be represented in the form

$$q = q_r + q_i\mathbf{i} + q_j\mathbf{j} + q_k\mathbf{k}, \quad (1)$$

where q_r , q_i , q_j , q_k , are real numbers, and \mathbf{i} , \mathbf{j} , \mathbf{k} are the basis elements of quaternions or, equivalently, the three imaginary units obeying the following rules

$$\begin{aligned} \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1, \quad \mathbf{i}\mathbf{j} = -\mathbf{j}\mathbf{i} = \mathbf{k}, \\ \mathbf{j}\mathbf{k} = -\mathbf{k}\mathbf{j} = \mathbf{i}, \quad \mathbf{k}\mathbf{i} = -\mathbf{i}\mathbf{k} = \mathbf{j}. \end{aligned} \quad (2)$$

The conjugate and modulus of a quaternion q are respectively defined by

$$q^* = q_r - q_i\mathbf{i} - q_j\mathbf{j} - q_k\mathbf{k}, \quad (3)$$

$$\|q\| = \sqrt{qq^*} = \sqrt{q_r^2 + q_i^2 + q_j^2 + q_k^2}. \quad (4)$$

Let $f(x, y)$ be an RGB image function. Each image pixel can be represented as a pure quaternion

$$f(x, y) = f_R(x, y)\mathbf{i} + f_G(x, y)\mathbf{j} + f_B(x, y)\mathbf{k}, \quad (5)$$

where $f_R(x, y)$, $f_G(x, y)$ and $f_B(x, y)$ are respectively the red, green and blue components of the pixel.

Using (5), a color image f is represented as a pure quaternion matrix. This representation has been successfully used in many fields of color image processing [4–16].

2.2. Quaternion discrete Fourier transform

Quaternion or hypercomplex Fourier transform has been independently introduced by Ell [19] and Bülow [20]. Sangwine [21] then applied it to digital color image processing and defined QDFT and its inverse transform IQDFT. Since the multiplication of the quaternions is not commutative due to the rules (2), Pei et al.

[22] presented three types of QDFT and IQDFT: two-side [21], left-side and right-side. The right-side one is used in this paper. Notice that the left-side one can be processed in a similar way. Right-side QDFT and IQDFT are defined as follows

$$F^R(u, v) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)}, \quad (6)$$

$$f(m, n) = F^{-R}(m, n) = \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^R(u, v) e^{\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)}, \quad (7)$$

where $f(m, n)$ is a color image of size $M \times N$ represented in the quaternion form as (5), μ is a unit pure quaternion. Basically, μ can be defined as a linear combination of \mathbf{i} , \mathbf{j} , and \mathbf{k} such as: $\mu = \alpha \mathbf{i} + \beta \mathbf{j} + \gamma \mathbf{k}$, $\alpha, \beta, \gamma \in \mathbf{R}$, $\|\mu\| = 1$.

3. Watermarking of QDFT coefficients

In this section, considering the general pure unit quaternion μ defined above, we first demonstrate the relationships that exist between the different components of their QDFT coefficients with the conventional DFT of the RGB color channels so as to expose the constraints to satisfy in order to avoid watermark energy loss. We then analyze the watermarking capacities one must expect when modulating QDFT components under these constraints regarding the different unit pure quaternions \mathbf{i} , \mathbf{j} , \mathbf{k} , μ_{PerC} and μ_{Lum} .

3.1. "Symmetry" constraints for watermarking energy preservation

It can be drawn from (7) that $F^{-R}(m, n)$ must be a pure quaternion matrix after some QDFT coefficients have been modified for watermark insertion. Otherwise, taking only the three imaginary parts of this quaternion matrix (see (5)) to get the watermarked image $f(m, n)$ will discard non-null real part data and result in a loss of watermark energy as previously mentioned (i.e. extracted and embedded watermarks will be different). In the sequel, we establish the "symmetry constraints" one must respect so as to avoid this issue.

The QDFT and IQDFT can be obtained from the conventional DFT and inverse discrete Fourier transform (IDFT) of each color channel as discussed below.

Considering the general unit pure quaternion $\mu = \alpha \mathbf{i} + \beta \mathbf{j} + \gamma \mathbf{k}$, $\alpha, \beta, \gamma \in \mathbf{R}$, substituting (5) into (6), we have

$$\begin{aligned} F^R(u, v) &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [f_R(m, n)\mathbf{i} + f_G(m, n)\mathbf{j} + f_B(m, n)\mathbf{k}] \\ &\quad \times e^{-\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &= \mathbf{i} \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_R(m, n) e^{-\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &\quad + \mathbf{j} \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_G(m, n) e^{-\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &\quad + \mathbf{k} \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_B(m, n) e^{-\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &= \mathbf{i} \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_R(m, n) \\ &\quad \times \left\{ \cos \left[2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right) \right] - \mu \sin \left[2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right) \right] \right\} \end{aligned}$$

$$\begin{aligned} &+ \mathbf{j} \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_G(m, n) \\ &\quad \times \left\{ \cos \left[2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right) \right] - \mu \sin \left[2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right) \right] \right\} \\ &+ \mathbf{k} \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_B(m, n) \\ &\quad \times \left\{ \cos \left[2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right) \right] - \mu \sin \left[2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right) \right] \right\} \\ &= \mathbf{i} [\text{Re}(DFT(f_R)(u, v)) + \mu \text{Im}(DFT(f_R)(u, v))] \\ &\quad + \mathbf{j} [\text{Re}(DFT(f_G)(u, v)) + \mu \text{Im}(DFT(f_G)(u, v))] \\ &\quad + \mathbf{k} [\text{Re}(DFT(f_B)(u, v)) + \mu \text{Im}(DFT(f_B)(u, v))] \\ &= A(u, v) + \mathbf{i}B(u, v) + \mathbf{j}C(u, v) + \mathbf{k}D(u, v), \end{aligned} \quad (8)$$

where

$$\begin{aligned} A(u, v) &= -\alpha \text{Im}(DFT(f_R)(u, v)) - \beta \text{Im}(DFT(f_G)(u, v)) \\ &\quad - \gamma \text{Im}(DFT(f_B)(u, v)), \\ B(u, v) &= \text{Re}(DFT(f_R)(u, v)) + \gamma \text{Im}(DFT(f_G)(u, v)) \\ &\quad - \beta \text{Im}(DFT(f_B)(u, v)), \\ C(u, v) &= \text{Re}(DFT(f_G)(u, v)) + \alpha \text{Im}(DFT(f_B)(u, v)) \\ &\quad - \gamma \text{Im}(DFT(f_R)(u, v)), \\ D(u, v) &= \text{Re}(DFT(f_B)(u, v)) + \beta \text{Im}(DFT(f_R)(u, v)) \\ &\quad - \alpha \text{Im}(DFT(f_G)(u, v)). \end{aligned} \quad (9)$$

Here $DFT(f_R)$, $DFT(f_G)$ and $DFT(f_B)$ are respectively the conventional DFT matrix of the red, green and blue channels, $\text{Re}(x)$ represents the real part of the conventional complex number x , and $\text{Im}(x)$ its imaginary part, that is, $\text{Re}(a + bi) = a$, $\text{Im}(a + bi) = b$. It can be found from (8) that the QDFT frequency domain is a 4-D space since each quaternion coefficient has four components.

Similarly, applying IQDFT to (8), we get the reconstructed image

$$\begin{aligned} \bar{f}(m, n) &= \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} [A(u, v) + \mathbf{i}B(u, v) + \mathbf{j}C(u, v) + \mathbf{k}D(u, v)] \\ &\quad \times e^{\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &= \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} A(u, v) e^{\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &\quad + \mathbf{i} \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} B(u, v) e^{\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &\quad + \mathbf{j} \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u, v) e^{\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &\quad + \mathbf{k} \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} D(u, v) e^{\mu 2\pi \left(\frac{mv}{M} + \frac{nu}{N} \right)} \\ &= \text{Re}(IDFT(A)(m, n)) + \mu \text{Im}(IDFT(A)(m, n)) \\ &\quad + \mathbf{i} [\text{Re}(IDFT(B)(m, n)) + \mu \text{Im}(IDFT(B)(m, n))] \\ &\quad + \mathbf{j} [\text{Re}(IDFT(C)(m, n)) + \mu \text{Im}(IDFT(C)(m, n))] \\ &\quad + \mathbf{k} [\text{Re}(IDFT(D)(m, n)) + \mu \text{Im}(IDFT(D)(m, n))] \\ &= \bar{f}_A(m, n) + \mathbf{i}\bar{f}_B(m, n) + \mathbf{j}\bar{f}_C(m, n) + \mathbf{k}\bar{f}_D(m, n), \end{aligned} \quad (10)$$

where

$$\begin{aligned}
\bar{f}_A(m, n) &= \text{Re}(\text{IDFT}(A)(m, n)) - \alpha \text{Im}(\text{IDFT}(B)(m, n)) \\
&\quad - \beta \text{Im}(\text{IDFT}(C)(m, n)) - \gamma \text{Im}(\text{IDFT}(D)(m, n)), \\
\bar{f}_R(m, n) &= \text{Re}(\text{IDFT}(B)(m, n)) + \alpha \text{Im}(\text{IDFT}(A)(m, n)) \\
&\quad + \gamma \text{Im}(\text{IDFT}(C)(m, n)) - \beta \text{Im}(\text{IDFT}(D)(m, n)), \\
\bar{f}_G(m, n) &= \text{Re}(\text{IDFT}(C)(m, n)) + \beta \text{Im}(\text{IDFT}(A)(m, n)) \\
&\quad + \alpha \text{Im}(\text{IDFT}(D)(m, n)) - \gamma \text{Im}(\text{IDFT}(B)(m, n)), \\
\bar{f}_B(m, n) &= \text{Re}(\text{IDFT}(D)(m, n)) + \gamma \text{Im}(\text{IDFT}(A)(m, n)) \\
&\quad + \beta \text{Im}(\text{IDFT}(B)(m, n)) - \alpha \text{Im}(\text{IDFT}(C)(m, n)).
\end{aligned} \tag{11}$$

Here $\text{IDFT}(\cdot)$ is the conventional inverse discrete Fourier transform.

Based on the above relationships (8) and (10) and depending on the pure unit quaternion considered, one can identify the symmetry constraints to respect when modifying QDFT coefficients so as to avoid watermark energy loss. After the watermark embedding process, \bar{f} should be a pure quaternion, or more clearly

$$\bar{f}_A = \mathbf{0}, \tag{12}$$

where $\mathbf{0}$ is a zero matrix. This corresponds to the precondition we discussed in the introduction.

In order to identify these symmetry constraints, we propose to distinguish three different types of pure unit quaternion μ .

(1) μ is a basis element (i.e. $\mu = \mathbf{i}$, $\mu = \mathbf{j}$ or $\mu = \mathbf{k}$)

We first consider $\mu = \mathbf{i}$. From (9), using the conjugate symmetry of DFT coefficients, we can easily obtain $A(0, 0) = 0$ and the following anti-symmetric and symmetric properties for the coefficient matrix A and B respectively.

$$\begin{aligned}
A(u, v) &= -A(M - u, N - v), & A(u, 0) &= -A(M - u, 0), \\
A(0, v) &= -A(0, N - v), \\
u &= 1, 2, \dots, M - 1, \quad v = 1, 2, \dots, N - 1.
\end{aligned} \tag{13}$$

$$\begin{aligned}
B(u, v) &= B(M - u, N - v), & B(u, 0) &= B(M - u, 0), \\
B(0, v) &= B(0, N - v), \\
u &= 1, 2, \dots, M - 1, \quad v = 1, 2, \dots, N - 1.
\end{aligned} \tag{14}$$

Eqs. (13) and (14) are the symmetry constraints to respect when $\mu = \mathbf{i}$. Indeed, if we modify $A(u, v)$ and $B(u, v)$ according to these anti-symmetry and symmetry rules, it can be shown from (11) that the precondition (12) is satisfied no matter how $C(u, v)$ and $D(u, v)$ are modified. Taking the three imaginary parts of \bar{f} as the three color channels of the watermarked image (i.e. $\mathbf{i}\bar{f}_R(m, n) + \mathbf{j}\bar{f}_G(m, n) + \mathbf{k}\bar{f}_B(m, n)$) does not lead to a watermark energy loss.

Results for $\mu = \mathbf{j}$ and $\mu = \mathbf{k}$ can be similarly derived.

If $\mu = \mathbf{j}$ in (8) and (10), it can be easily shown that B and D are free of constraints but not A and C . A is constrained by (13) like for $\mu = \mathbf{i}$, while C is constrained in the following way

$$\begin{aligned}
C(u, v) &= C(M - u, N - v), & C(u, 0) &= C(M - u, 0), \\
C(0, v) &= C(0, N - v), \\
u &= 1, 2, \dots, M - 1, \quad v = 1, 2, \dots, N - 1.
\end{aligned} \tag{15}$$

And last, if $\mu = \mathbf{k}$, B and C are free of constraints. The constraint of A is the same as for $\mu = \mathbf{j}$ and $\mu = \mathbf{i}$: A should verify (13), while that of D is given by

$$\begin{aligned}
D(u, v) &= D(M - u, N - v), & D(u, 0) &= D(M - u, 0), \\
D(0, v) &= D(0, N - v), \\
u &= 1, 2, \dots, M - 1, \quad v = 1, 2, \dots, N - 1.
\end{aligned} \tag{16}$$

As a conclusion, for $\mu = \mathbf{i}$, \mathbf{j} , or \mathbf{k} , as we will see in Section 3.2 and Section 5, it becomes possible to insert a watermark into the four components at the same time so as to take full advantage of the 4-D QDFT domain.

(2) μ is a linear combination of two basis elements (e.g. $\mu = \mu_{\text{Perc}}$)

In this case, only the component A has to respect the anti-symmetry constraint shown in (13). Moreover, it can be seen from (11) that, due to the fact that one of the three numbers α , β and γ is equal to 0, the corresponding component B , C or D is free of constraints. Whatever how this component is modified the precondition (12) is verified.

To illustrate this comment, let us consider $\mu_{\text{Perc}} = (-2\mathbf{j} + 8\mathbf{k})/\sqrt{68}$. Bas et al. first used it in [8] where they insert only one bit per 8×8 pixel block using QIM without considering the precondition (12) [9,13]. By substituting $\mu = \mu_{\text{Perc}}$ in (8) and (10), we can easily derive the same anti-symmetry property of A given by (13) as well as

$$\bar{f}_A = \text{Re}(\text{IDFT}(A)) + \frac{2 \text{Im}(\text{IDFT}(C)) - 8 \text{Im}(\text{IDFT}(D))}{\sqrt{68}}. \tag{17}$$

In order to ensure the precondition (12), Eq. (17) should be equal to 0. As a consequence, A cannot be modified without respecting (13). Herein, B is free of constraints. It can be modified; the precondition (12) will be respected. So, the scheme of Bas et al. can be improved by conducting watermark embedding in $A(u, v)$ according to (13) and in $B(u, v)$ rather than in $C(u, v)$ and $D(u, v)$. Notice that we have considered this improvement in our experiments in Section 5.

(3) μ is a linear combination of three basis elements (e.g. $\mu = \mu_{\text{Lum}}$)

For this last case, only A constrained by (13). Let us take $\mu_{\text{Lum}} = (\mathbf{i} + \mathbf{j} + \mathbf{k})/\sqrt{3}$ adopted in [9,13,16] as example. It makes A having the same anti-symmetry property as in (13) and

$$\begin{aligned}
\bar{f}_A &= \text{Re}(\text{IDFT}(A)) - \text{Im}(\text{IDFT}(B))/\sqrt{3} - \text{Im}(\text{IDFT}(C))/\sqrt{3} \\
&\quad - \text{Im}(\text{IDFT}(D))/\sqrt{3}.
\end{aligned} \tag{18}$$

In order to preserve the precondition (12), B , C and D cannot be modified. A can be but under the constraint (13).

To conclude, with μ_{Perc} and μ_{Lum} , one can only embed a watermark into two or one QDFT component while satisfying $\bar{f}_A = \mathbf{0}$.

3.2. Watermarking capacities under symmetry constraints

Watermarking capacity is by definition related to the number of QDFT components one can modulate independently. The greater this number, the greater the capacity is. As explained above, the choice of the unit pure quaternion impacts the number of components available for watermark embedding. Indeed, for a given unit pure quaternion, there is a set of symmetry constraints to respect so as to avoid watermark energy loss. Notice that this choice impacts not only algorithms' performance in terms of capacity, as we will see in this section, but also their performance in terms of invisibility and robustness as we will see in Section 5.

Let us consider QDFT coefficients of 8×8 pixel blocks. Let us also assume that only a part of them at some medium frequency positions, as given in Fig. 1, are secretly selected as candidate for embedding and that QDFT coefficient components (i.e. $A(u, v)$, $B(u, v)$, $C(u, v)$ or $D(u, v)$) are modulated by adding or subtracting the real value Δ . We summarize in Fig. 2 the QDFT components that are available for watermark embedding for each of the

			1	9	10	22	23
			2	8	11	21	37
	3	7	12	20	25	36	38
4	6	13	19	26	35	39	48
5	14	18	27	34	40	47	49
15	17	28	33	41	46	50	
16	29	32	42	45	51		
30	31	43	44	52			

Fig. 1. QDFT medium frequency positions we retain as candidates for embedding while considering 8×8 pixel blocks.

unit pure quaternion, i, j, k, μ_{Perc} and μ_{Lum} , while respecting the aforementioned symmetry constraints and considering the whole candidate set in Fig. 1.

If we look at the quaternion $\mu = i$, one can independently modulate QDFT components marked in gray in Fig. 2 while, if necessary, modifying (anti)-symmetrically QDFT components indicated in black based on (13) and (14). It is important to notice that QDFT components indicated in hatched dark have to be modified accordingly (13) and (14) even if they are not candidates for watermark embedding (see Fig. 1). Otherwise, a watermark energy loss will occur. To sum up, within the set of $208 = 4 \times 52$ QDFT compo-

Table 1

Capacities of different algorithms per 8×8 pixel block embedding one bit into one coefficient.

Algorithms	μ_{Lum} [9]	μ_{Perc} [8]	Our i, j or k
Capacity (bits)	32	84	168

nent candidates, 168 can be modulated independently (64 from A and B , 104 from C and D). We achieve the same capacity performance for $\mu = j$ and $\mu = k$ based on the same principles, i.e. by considering (13) and (15) for $\mu = j$ and (13) and (16) for $\mu = k$.

If we focus on $\mu = \mu_{Perc}$, only A and B components can be exploited. Modulation of $A(u, v)$ components is “symmetry” constrained by (13). Herein, there are 32 components from A and 52 from B that can be modulated independently (see Fig. 2). In the case $\mu = \mu_{Lum}$, the single component A is used under the constraints given by (13). As a consequence, there are only 32 components usable for embedding.

These “capacities” are summarized in Table 1. This latter shows that one can achieve with $\mu = i, j, k$ a capacity twice and five times greater than with $\mu = \mu_{Perc}$ and μ_{Lum} , respectively, under the constraint of no watermark energy loss. In Section 5, we further experimentally demonstrate that we can also achieve with $\mu = i, j, k$ better performance in terms of watermark imperceptibility and robustness. Indeed, the watermark is spread over the whole color space.

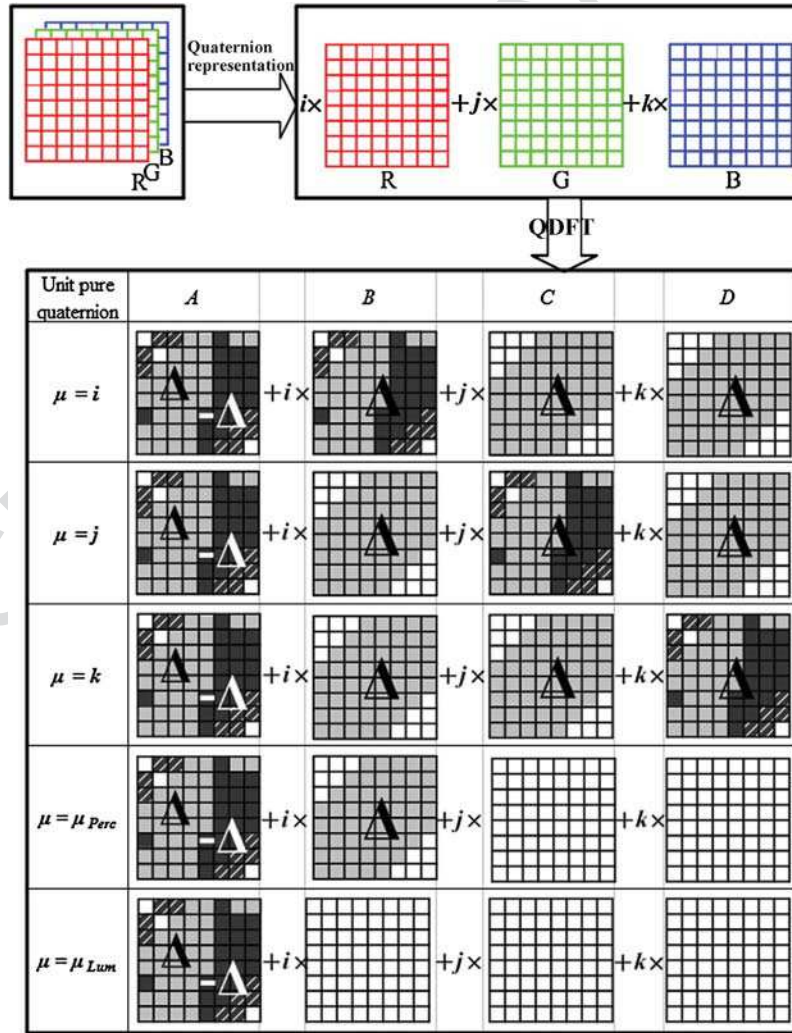


Fig. 2. Available QDFT coefficients for watermark embedding (Δ is the QDFT component distortion).

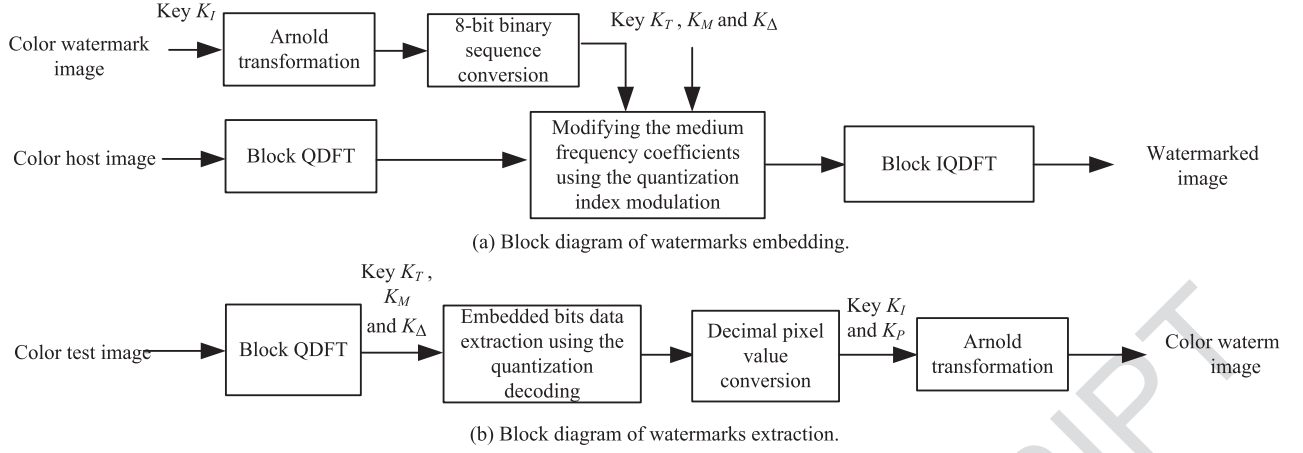


Fig. 3. Block diagram of the proposed color watermarking framework.

4. Experimental watermarking framework

In order to illustrate the gain of performance we achieve when considering the four dimensions of the QDFT domain holistically compared to other strategies, we propose to consider a general watermarking framework where our solution and the modulations of Jiang et al. [9], and Wang et al. [16] as well as the improved version of the one of Bas et al. [8] are implemented and tested for fair comparison. We describe thereafter the embedding and extraction procedure of this experimental framework before presenting in Section 4.3 the different algorithms we derived from it.

4.1. Watermarks embedding

The block diagram of the color watermarking scheme we implemented is shown in Fig. 3. As in [8,9,16], QDFT coefficient components are modulated by means of the well-known QIM [17]. However, unlike [8,9,16], we added a pre-processing stage which includes a watermark Arnold transformation followed by a conversion of the scrambled watermark into a stream of bits. The former purpose is to enhance the security and the robustness of the watermark to cropping, while the purpose of the latter is to make the scheme more robust against common image processing attacks (such as filtering, JPEG compression, and image blurring). In this scheme, compared to [8,9,13,16], we also have considered the embedding of color and intelligible watermarks. This is based on the facts that in some specific applications, embedded data are required to be intelligible and chromatic (e.g. strings of characters, logos or signature, etc.). Besides, extracted chromatic watermarks are easier to identify visually than achromatic watermarks (see Section 5).

Thus, let w be a color watermark. It is embedded K_T times into a host image h using a multiple redundant embedding strategy [23,24] through the following steps:

- (1) Watermark image pre-processing – In order to enhance the security and the robustness to cropping, w is scrambled by means of an Arnold transformation of period K_P and K_I iterations. Each pixel value of the scrambled watermark is then converted into an 8-bit sequence so as to finally get a bitstream bw .
- (2) Host image blocking – h is divided into 8×8 pixel blocks and the QDFT of each is computed.
- (3) QDFT coefficient modulation – For each block, the QDFT coefficient candidates shown in Fig. 1 are secretly selected for embedding. Depending on the algorithm exploited (i.e. our or [8,9,13,16]), for each retained position (u, v) , one to four of the

QDFT coefficient components (i.e. $A_h(u, v)$, $B_h(u, v)$, $C_h(u, v)$ and $D_h(u, v)$) are modified. One bit of bw is inserted using the QIM as follows

$$I'_h(u, v) = \begin{cases} (l + \frac{1}{2})K_\Delta, & \text{if } bw(x) \neq l \pmod{2} \\ (l - \frac{1}{2})K_\Delta, & \text{if } bw(x) = l \pmod{2} \end{cases} \quad (19)$$

where $l = \lfloor \frac{I_h(u, v)}{K_\Delta} \rfloor$, $\lfloor \cdot \rfloor$ is the rounding function, $I_h(u, v)$ is one of four components, $I'_h(u, v)$ the modified component value, x is the coordinate of the watermark bit in bw , K_Δ is the quantization step, and mod is the modulo operator.

QDFT coefficients are symmetrically modified according to the properties we established in Section 3 for different quaternions.

- (4) Inverse QDFT – The IQDFT is applied to obtain the watermarked image. Notice that the watermark bitstream bw is embedded repeatedly K_T times.

The private key of our system is constituted of: the number of times w is repeated (K_T), the period of the Arnold Transform (K_P) and the number of iteration (K_I), the quantization step (K_Δ), as well as a key K_M that indicates which medium frequency QDFT coefficients are used for embedding.

4.2. Watermarks extraction

Based on the private key, the color watermark is extracted as follows:

- (1) Divide the test image into 8×8 pixel blocks and compute their QDFT coefficients.
- (2) Extract K_T bitstream watermarks bw^s , $s = 1, 2, \dots, K_T$, from the frequencies that have been retained for embedding in each block

$$bw^s(x) = \left\lfloor \frac{I'_h(u, v)}{K_\Delta} \right\rfloor \pmod{2}, \quad (20)$$

where $I'_h(u, v)$ is one watermarked component of a QDFT coefficient and K_Δ the QIM quantization step.

Then, achieve the watermark bitstream bw' by using

$$bw'(x) = \begin{cases} 1, & \overline{bw}(x) \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (21)$$

where $\overline{bw}(x)$ is the mean of the values of $bw^s(x)$, $s = 1, 2, \dots, K_T$.

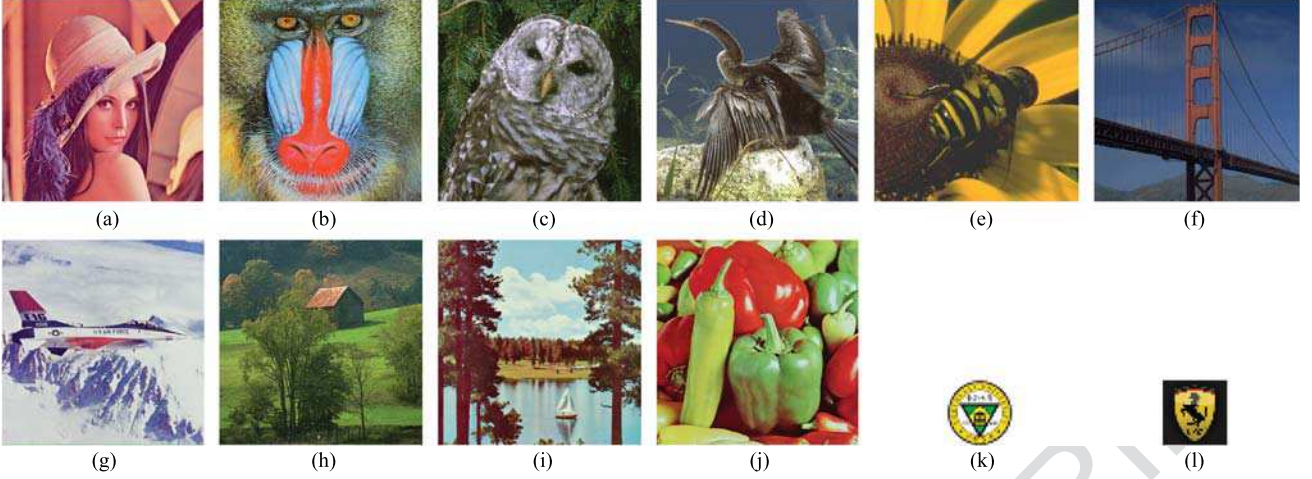


Fig. 4. (a)–(j) Examples of color host images of 512×512 pixels issued from the image database of the Computer Vision Group at the University of Granada; (k)–(l) watermark images of 32×32 .

- (3) Bits bw' are partitioned into 8-bit groups and converted into pixel values or re-organized into a 2D color image. This extracted scrambled watermark is carried out $(K_P - K_I)$ Arnold transforms again so as to obtain the final color watermark image w' .

Notice that because the watermark extraction does not need the original host image and the original watermark, the proposed algorithm is blind.

4.3. Derived watermarking schemes

Depending on the unit pure quaternion considered in the previous framework, different schemes can be derived. Indeed, if μ_{lum} is used, we retrieve the common basis of the algorithms described in [9,13,16] herein combined with the pre-processing stage. It is the same if μ_{perc} is considered. We obtain a scheme similar to the method of Bas et al. [8]. Our schemes correspond to cases where i , j or k are considered. Thus the previous framework can be derived into five distinct schemes.

In the following section, we compare these five different approaches in terms of capacity, invisibility and robustness against different kind of attacks.

5. Experimental results and analysis

In order to evaluate the efficiency of the previous schemes, experiments have been conducted in terms of invisibility, capacity and robustness to different kinds of image modifications. We have also compared these five schemes with the one proposed by Su et al. [3]. This one treats RGB color channels independently, and modulates two coefficients of Schur decomposition of 4×4 pixel block so as to insert one bit. Herein, for a fair comparison, Su's method has been extended so as to follow the same watermark preprocessing step. It now includes a multiple redundant watermark embedding strategy (see Section 4.1). As a consequence, all compared algorithms are based on the same framework.

5.1. Measures of performance

Two objective criteria have been considered:

- (1) The Peak Signal to Noise Ratio (PSNR) is used to evaluate the invisibility of the embedded watermark. Considering the host color image h of size $M_h \times N_h$ and its watermarked version h' , the PSNR is defined as

$$\text{PSNR} = 10 \lg \left[\frac{255^2}{\frac{1}{3M_h N_h} \sum_{x=1}^{M_h} \sum_{y=1}^{N_h} \sum_{\xi \in \{R, G, B\}} (h_\xi(x, y) - h'_\xi(x, y))^2} \right], \quad (22)$$

where h_R, h_G, h_B and h'_R, h'_G, h'_B are the three color channels of h and h' , respectively.

- (2) The Normalized Correlation (NC) is here employed to measure the quality of the extracted color watermark image w' compared to its original version w , both of $M_w \times N_w$ pixels. The NC between $w = \{w_R, w_G, w_B\}$ and $w' = \{w'_R, w'_G, w'_B\}$ is given by

$$\text{NC} = \frac{\sum_{x=1}^{M_w} \sum_{y=1}^{N_w} \sum_{\xi \in \{R, G, B\}} w_\xi(x, y) w'_\xi(x, y)}{\sqrt{\sum_{x=1}^{M_w} \sum_{y=1}^{N_w} \sum_{\xi \in \{R, G, B\}} w_\xi^2(x, y)} \sqrt{\sum_{x=1}^{M_w} \sum_{y=1}^{N_w} \sum_{\xi \in \{R, G, B\}} w'^2_\xi(x, y)}}. \quad (23)$$

5.2. Tests of invisibility

To conduct these experiments, 109 color images available from the Computer Vision Group at the University of Granada¹ were considered. This dataset includes 83 images of 512×512 pixels (see samples in Fig. 4(a–j)), 15 images of size 128×128 and 11 of 256×256 pixels. We also have used two 32×32 color watermarks: the Southeast University logo (Fig. 4(k)) and the Ferrari logo (Fig. 4(l)).

Because the scheme proposed by Su et al. [3] can only embed one bit per 4×4 pixel block for one of the three RGB color channel, the maximum number of times a 32×32 color watermark can be repeated into a 512×512 color host image is two. For fair comparison, the number of times watermark are repeated was set to two, i.e. $K_T = 2$ (see Section 4.1), for all of tested schemes. Images of size 128×128 and 256×256 were also rescaled to 512×512 for comparison. It is important to notice that quaternion based methods we study offer better performance in terms of capacity. If we consider our three schemes based on i , j or k , such a 32×32 watermark can be repeatedly embedded more than 28 times if all QDFT coefficient candidates in Fig. 1 are considered for embedding (see Sections 3.2 and 4.1). Other parameters such as the number of iterations of the Arnold Transform, the quantization step of the QIM were set to $K_I = 10$ and $K_\Delta = 160$, respectively.

Table 2 provides PSNR values we achieved in average for these six different algorithms. Fig. 5 shows watermarked images when inserting Fig. 4(k) into the test Lena image Fig. 4(a) with different

¹ <http://decsai.ugr.es/cvg/dbimagenes/>.













Algorithms	Su et al. [3]	μ_{Lum} [9]	μ_{Perc} [8]
Watermarked image and its PSNR(dB)	 36.801	 36.400	 36.994
Extracted watermark and its NC	 0.997	 1.000	 1.000
Algorithms	Our <i>i</i>	Our <i>j</i>	Our <i>k</i>
Watermarked image and its PSNR(dB)	 37.717	 37.200	 37.666
Extracted watermark and its NC	 1.000	 1.000	 1.000

Fig. 5. Lena watermarked images obtained with the different embedding algorithms and their extracted watermarks. NC corresponds to the normalized correlation between the original and the extracted watermarks.

Table 2

PSNR between the host image and the watermarked image using different algorithms, host images H and watermark images W (AVE is the average PSNR of image set, STD is standard deviation).

Algorithms	H : Fig. 4(a)		H : Image set	
	W : Fig. 4(k)	W : Fig. 4(l)	W : Fig. 4(k)	W : Fig. 4(l)
Su et al. [3]	36.801	36.781	AVE: 34.789 STD: 4.371	AVE: 34.786 STD: 4.375
μ_{Lum} [9]	36.400	36.401	AVE: 36.477 STD: 0.342	AVE: 36.477 STD: 0.340
μ_{Perc} [8]	36.994	36.992	AVE: 37.025 STD: 0.309	AVE: 37.024 STD: 0.308
Our <i>i</i>	37.717	37.689	AVE: 37.679 STD: 0.291	AVE: 37.677 STD: 0.297
Our <i>j</i>	37.200	37.189	AVE: 37.258 STD: 0.302	AVE: 37.255 STD: 0.301
Our <i>k</i>	37.666	37.628	AVE: 37.675 STD: 0.295	AVE: 37.679 STD: 0.293

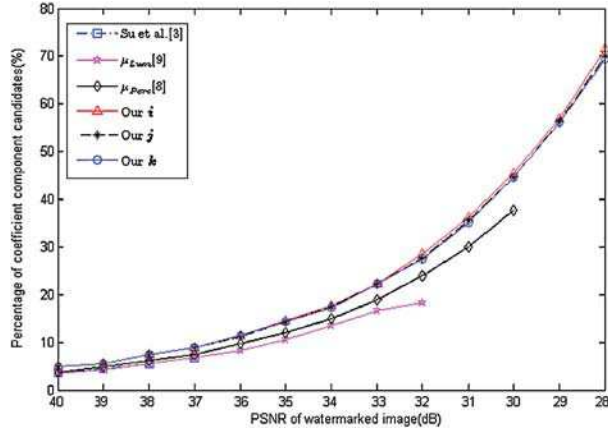
algorithms. Fig. 5 also gives a view of the extracted color watermarks and their corresponding NC values. It can be seen from Table 2 and Fig. 5 that our approach, based on *i*, *j* or *k*, better preserve the image quality than the other two quaternion-based algorithms. The reason mainly stands in the fact our approach use the four components of QDFT coefficients while the others not. Indeed, these latter have to modify more QDFT coefficients in each block so as to achieve the same number of watermark repetition (K_T). As stated in Section 3.2, our schemes' capacities are five times greater than those based on μ_{Lum} [9,13,16], and twice greater than the one with μ_{Perc} [8]. Notice that for all experiments, QDFT coefficients are successively watermarked according to the order indicated in Fig. 1. It can be seen that the more frequency coefficients we take for embedding, the more the image is distorted.

The above results show that the performance achieved by our schemes with the different unit pure quaternions *i*, *j* or *k* is close to each other. However, the choice of *i* behaves slightly better. Re-

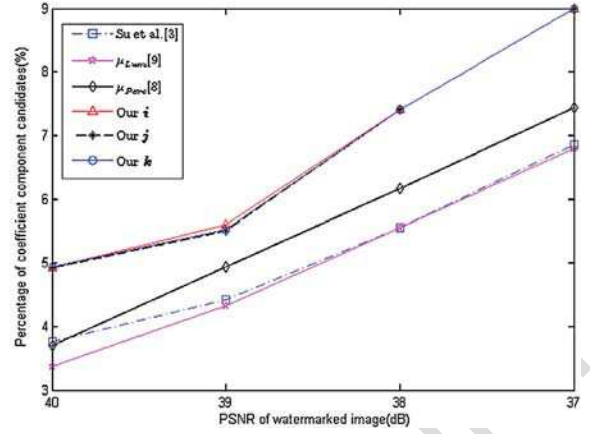
garding the other two quaternions, μ_{Perc} offers somewhat better performance than μ_{Lum} . This is in agreement with the observations made by Bas et al. in [8]. It can also be noticed that almost all quaternion-based algorithms are superior to the "more conventional" algorithm proposed by Su et al. [3]. This is due to the holistic processing advantage the quaternion representation of color images offers.

From Fig. 5, one can remark that color watermarks are extracted exactly after the embedding process except with the method of Su et al. (NC = 0.997). The reason is that the embedding into the Schur decomposition matrix impacts the orthogonality of this matrix.

In a second experiment, we focused on the performance in terms of capacity of each of the previous schemes for the same PSNR value. To do so, in order to reach one given PSNR value, each scheme was parameterized so as to modify QDFT coefficient components (A , B , C and D) of same position (see Fig. 1) before using the components of the QDFT coefficient in the next position. More clearly, considering $\mu = i$ as example, components $A(k)$, $B(k)$, $C(k)$ and $D(k)$ (where k is the k th position in Fig. 1) have to be modulated before $A(k+1)$ can be modified. In fact, we play with the number of times the watermark is repeated, i.e. K_T , which is a real value in that case. Obviously, all previous symmetry constraints are still considered. In the case the Lena test image (Fig. 4(a)) or equivalently all of its $(512 \times 512)/(8 \times 8) \times 168 = 688128$ QDFT coefficient component candidates (see Fig. 1), Fig. 6 depicts the percentage of these components used by each algorithm for the same PSNR values within the range 28 dB to 40 dB. It can be seen that: (1) capacities offered by the three schemes based on *i*, *j* and *k* are greater than those provided by the three others. This demonstrates the interest of spreading the watermark over the four dimensions of the QDFT domain; (2) schemes based on μ_{Lum} and μ_{Perc} cannot go below PSNR values of 32 dB and 30 dB respectively. This is due to the fact they can modulate only 19% and 50% of available QDFT coefficient components.

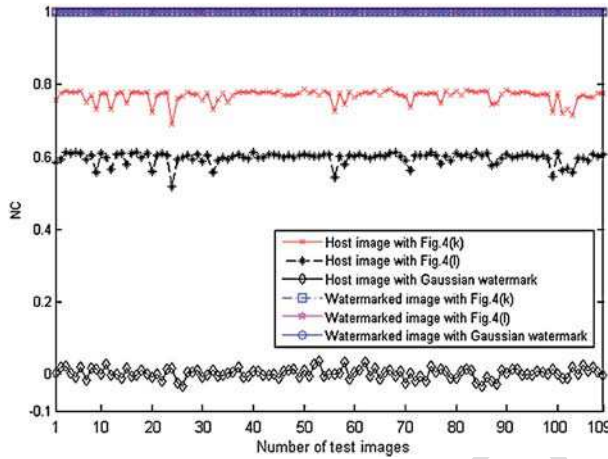


(a) PSNR from 40dB to 28dB

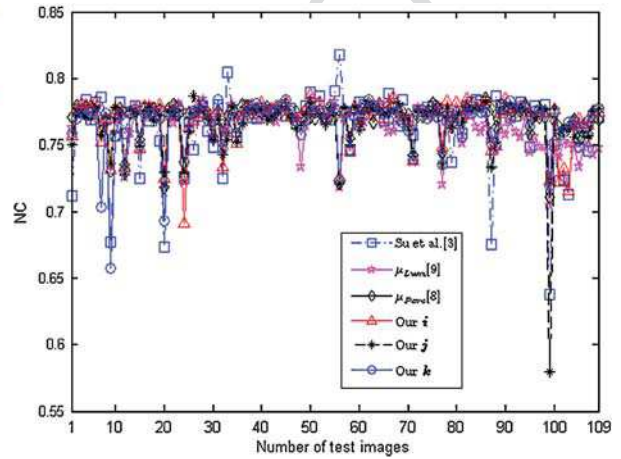


(b) PSNR from 40dB to 37dB (lower left corner of (a))

Fig. 6. Percentage of coefficient component candidates under the same PSNR value using different algorithms.



(a) Results for non-watermarked and watermarked images using our *i*



(b) Results for Fig. 4(k) and non-watermarked images using different algorithms

Fig. 7. NC values for color watermarks extracted from non-watermarked images and watermarked images.

5.3. Tests of robustness

5.3.1. Watermark detection and variability of NC values

Before evaluating the robustness of these six different schemes against various attacks (e.g. JPEG and salt and pepper noise), we first propose to verify that watermarked images have NC values fairly different from those obtained with non-watermarked images or images tattooed with a different watermark. To do so, each of the previous 109 color images were watermarked with three distinct watermarks: the Southeast University logo Fig. 4(k), the Ferrari logo Fig. 4(l) and a random Gaussian watermark. For all six compared algorithms, the corresponding NC values were computed between original watermarks and the ones extracted from the 436 watermarked and original images. Due to the space limitation and because all six algorithms have a similar behavior, we only provide in Fig. 7(a) the NC values we obtained with our algorithm using *i* considering the three previous watermarks and no-image-attack. As it can be seen there is a clear cut in between NC values of watermarked image (NC = 1.0) and non-watermarked images (i.e. original images or images watermarked with another watermark). It can also be observed that NC values of non-watermarked images depend on the watermark content and that this one does not vary too much whatever the host image. In fact each watermark has its own NC limit value. Herein, the smallest one is achieved with the Gaussian watermark and the highest one with the Southeast Uni-

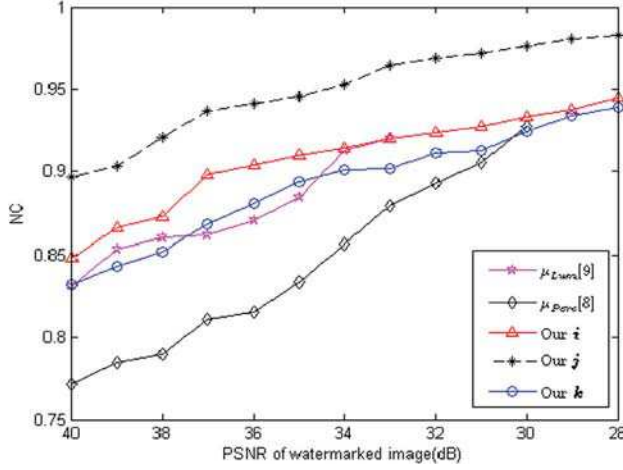
versity logo. This impacts the watermark detection process, that is to say if a specific watermark is present or not in an image. Indeed, for a given watermark, the closest the NC value is to this limit, the less we can claim the watermark is present in the image. It can be also remarked from Fig. 7(b) that for a given watermark all six algorithms achieve quite the same NC limit in average but with more or less variations. In the next subsections, where we evaluate algorithms' robustness, we will consider that the following maximum NC values extracted from Fig. 7(b): 0.787, 0.787, 0.784, 0.785, 0.784, and 0.818, are the Southeast University logo NC limits of our algorithms with *i*, *j*, *k*, μ_{Perc} and μ_{Lum} , and the algorithm of Su et al., respectively. NC limits for which it is impossible to claim an image has been watermarked with the Southeast University logo.

5.3.2. Evaluation of algorithms' robustness

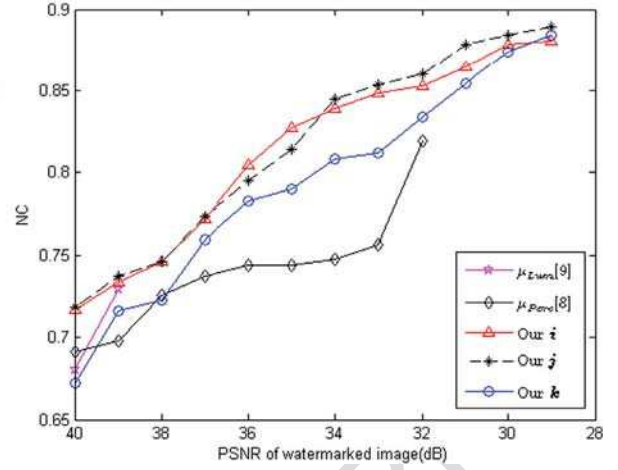
Without loss of generality, the 109 watermarked images obtained previously in the first experiment of Section 5.2 (with Fig. 4(k) as color watermark) are re-used in the following tests of robustness to various image modifications (e.g. JPEG compression, noise adding, cropping, filtering and geometric attack and so on).

(1) Robustness to the JPEG compression attack

We give in Fig. 9 and in average NC variations for the six tested algorithms considering the JPEG compression attack with a quality factor varying in the range [30, 90]. This result illustrates in general the performance of these schemes over our whole image test set.



(a) Embedding respecting symmetry constraints



(b) Embedding without respecting symmetry constraints

Fig. 8. NC value for color watermarks after JPEG compression attack with the quality factor $q = 60$.

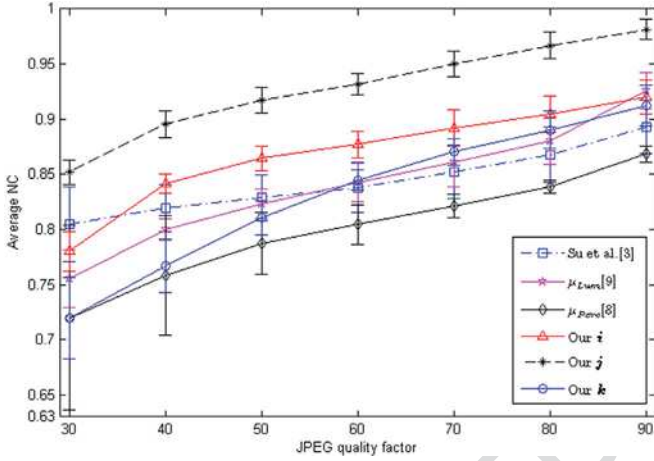


Fig. 9. NC values in average and standard deviation for color watermarks extracted from our image test set after a JPEG compression of different quality factors.

In order to better show their efficiency, watermarks extracted from the Lena test image attacked with a JPEG compression of quality factor 60 are displayed in Fig. 10. From these different figures, it can be seen that each algorithm tends to its NC limit (see Section 5.3.1) when the JPEG quality factor decreases. It can also be observed that our algorithms with i and j and especially the one based on j are much more robust than the others that reach their respective NC limit for quality factors of about 40 for the algorithm with μ_{Lum} and Su's algorithm, 50 for the algorithm using k and μ_{Perc} . Notice that for Su's algorithm, the obtained NC values are quite close to the NC limit when the quality factor is smaller than 70. Our algorithms with i and j survive a lossy compression of quality factor 40. The reason why j offers better performance may stand on the fact that j is assigned to the green color in the quaternion representation of color images; green color which carries most of the visual image information. Although when the quality factor is such as $q = 60$ (see Fig. 10), the watermark we extract using our algorithms is degraded with some noise but can be easily identified. This is not the case for the others. It shows also the interest for using intelligible color watermarks. Besides, as illustrated by the NC standard deviation, all algorithms are quite stable even the one of Su et al.

(2) Robustness to the noise attack

Gaussian noise attack of zero mean with variable standard deviation (STD) as well as salt-and-pepper noise attack with various

density were added to 109 watermarked test images. NC values of extracted color watermarks are shown in Fig. 11. They are given in average accompanied with their standard deviations. Notice that for presentation clarity, NC standard deviation results for the salt-and-pepper noise attack are presented in Table 3. Some extracted watermark samples and their corresponding NC values are given in Fig. 10. Again, each algorithm converges to its NC limit with the increase of the attack strength. These results indicate our algorithms with i , j or k are more robust than the others to noise addition. Moreover, the algorithm with j is also the best one. Besides, all quaternion-based algorithms are also superior to the one of Su et al. This once again proves the advantage of quaternion-based algorithms.

(3) Robustness to the cropping attack

Cropping attack was simulated by setting the left part of image to white pixel value $255i + 255j + 255k$. Our watermarked images were then cropped with different proportions varying from 5% to 60%. Obtained NC values are shown in Fig. 12. Some extracted watermarks are also provided in Fig. 10. These two figures show that the five quaternion-based algorithms are very robust to cropping either in terms of NC value or visually. It can be observed from Fig. 10 that, even though test images are 60% cropped, the watermarks extracted with five quaternion-based algorithms can still be identified. This is not the case for the algorithm of Su et al. Moreover, the five quaternion-based algorithms are also very stable as shown by their very small standard deviation.

(4) Robustness to the filtering attack

In this experiment, the 109 watermarked images were filtered by various types of filtering with a mask size of 3×3 pixels. Algorithms' results are given in Fig. 10. They show that our algorithms are more robust than the others. Extracted watermarks using the algorithm of Su et al. are difficult to recognize except for the motion blurring attack.

(5) Robustness to the geometric attack

Because our watermarking framework has not been elaborated in order to be robust to geometric attacks, we turn to test the robustness of the tested algorithms to the truncation errors that result from the correction of geometric attack. More clearly, in this experiment, one watermarked test image is first carried out a forward geometric transformation, and is then corrected by an inverse transformation. Scaling and rotation have been considered here as geometric attacks. Nowadays, different approaches [25,26] have been proposed so has to determine the parameters of the geometric transform the image undergone. As example, in the more general case of affine transformations, Zhang et al. [26] compare





























































Attacks		Su <i>et al.</i> [3]	μ_{Lum} [9]	μ_{Perc} [8]	Our <i>i</i>	Our <i>j</i>	Our <i>k</i>
JPEG compression with quality factor $q = 60$	Lena test image	 0.852	 0.865	 0.814	 0.881	 0.932	 0.854
	Image test set	AVE:0.838 STD: 0.023	AVE:0.843 STD: 0.018	AVE:0.804 STD: 0.018	AVE: 0.877 STD: 0.015	AVE: 0.932 STD: 0.010	AVE: 0.845 STD: 0.017
Gaussian noise with STD $\sigma = 10$	Lena test image	 0.900	 0.911	 0.933	 0.934	 0.949	 0.933
	Image test set	AVE:0.886 STD: 0.030	AVE:0.908 STD: 0.003	AVE:0.935 STD: 0.003	AVE: 0.935 STD: 0.002	AVE: 0.948 STD: 0.002	AVE: 0.935 STD: 0.003
Salt-and-pepper noise with density $p = 1\%$	Lena test image	 0.885	 0.904	 0.914	 0.903	 0.916	 0.907
	Image test set	AVE:0.885 STD: 0.014	AVE:0.899 STD: 0.016	AVE:0.916 STD: 0.008	AVE: 0.912 STD: 0.012	AVE: 0.919 STD: 0.011	AVE: 0.913 STD: 0.012
Cropping with the proportion 60%	Lena test image	 0.876	 0.898	 0.899	 0.898	 0.898	 0.899
	Image test set	AVE:0.872 STD: 0.004	AVE:0.897 STD: 0.002	AVE:0.898 STD: 0.002	AVE: 0.897 STD: 0.002	AVE: 0.898 STD: 0.002	AVE: 0.897 STD: 0.002
Gaussian filter (3×3)	Lena test image	 0.895	 0.919	 0.956	 0.966	 0.963	 0.966
	Image test set	AVE:0.865 STD: 0.049	AVE:0.912 STD: 0.039	AVE:0.952 STD: 0.030	AVE: 0.958 STD: 0.032	AVE: 0.968 STD: 0.027	AVE: 0.958 STD: 0.032
Median filter (3×3)	Lena test image	 0.821	 0.871	 0.925	 0.960	 0.955	 0.953
	Image test set	AVE:0.800 STD: 0.030	AVE:0.859 STD: 0.021	AVE:0.914 STD: 0.031	AVE: 0.945 STD: 0.038	AVE: 0.948 STD: 0.035	AVE: 0.945 STD: 0.038
Average filter (3×3)	Lena test image	 0.816	 0.856	 0.906	 0.944	 0.943	 0.943
	Image test set	AVE:0.799 STD: 0.026	AVE:0.855 STD: 0.023	AVE:0.903 STD: 0.025	AVE: 0.936 STD: 0.033	AVE: 0.940 STD: 0.032	AVE: 0.937 STD: 0.033
Motion blurring (3×3)	Lena test image	 0.939	 0.942	 0.964	 0.988	 0.987	 0.987
	Image test set	AVE:0.958 STD: 0.038	AVE:0.937 STD: 0.017	AVE:0.961 STD: 0.015	AVE: 0.984 STD: 0.017	AVE: 0.986 STD: 0.015	AVE: 0.984 STD: 0.017
Rotation with angle $\theta = 30^\circ$	Lena test image	 0.920	 0.942	 0.960	 0.971	 0.977	 0.967
	Image test set	AVE:0.900 STD: 0.052	AVE:0.935 STD: 0.044	AVE:0.950 STD: 0.037	AVE: 0.960 STD: 0.036	AVE: 0.963 STD: 0.033	AVE: 0.960 STD: 0.036
Scaling with factor $\lambda = 60\%$	Lena test image	 0.886	 0.903	 0.906	 0.928	 0.926	 0.922
	Image test set	AVE:0.860 STD: 0.046	AVE:0.890 STD: 0.043	AVE:0.904 STD: 0.041	AVE: 0.923 STD: 0.042	AVE: 0.927 STD: 0.040	AVE: 0.922 STD: 0.041

Fig. 10. NC values for color watermarks extracted from Lena test image and our image test set in average and standard deviation after various kinds of attacks (AVE is the average NC values of image test set, STD is standard deviation).

the Legendre moments of the unmodified watermarked image with those of the attack one so as to estimate the parameter of the affine transform which can then be reversed.

If we come back to our experiment, the 109 watermarked images were rotated by different angles from 10° to 90° . The NC values for these rotated test images are given in average in Fig. 13 while Table 4 provides the corresponding standard deviations. Some of the watermarks extracted from the Lena test image after attack are also displayed in Fig. 10. It can be observed that

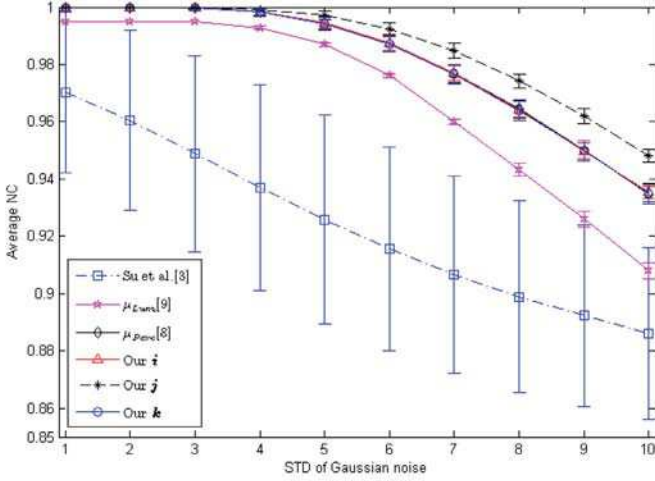
our algorithms (in particular with *i* and *j*) are also more robust than the others, including the two quaternion methods based on μ_{Perc} and μ_{Lum} . Beyond and as shown by their small standard deviations in Table 4 all tested algorithms have a stable behavior.

Watermarked images were also transformed using different scaling factors varying from 20% to 200%. Results in terms of NC are shown in average and standard deviation in Fig. 14 and Table 5 respectively, while some extracted watermark samples are given in Fig. 10. In general, the five quaternion-based algorithms are more

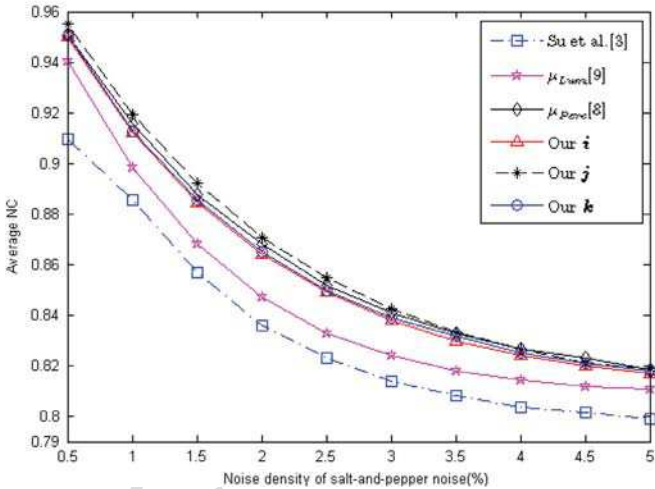
Table 3

NC standard deviation for color watermarks extracted from the image test set after adding salt-and-pepper noise attacks of different densities.

Salt-and-pepper noise attack with density ρ	Su et al. [3]	μ_{lum} [9]	μ_{perc} [8]	Our i	Our j	Our k
$\rho = 0.5\%$	0.0015	0.013	0.006	0.007	0.006	0.007
$\rho = 1.0\%$	0.0014	0.016	0.008	0.012	0.011	0.012
$\rho = 1.5\%$	0.0014	0.015	0.009	0.014	0.013	0.014
$\rho = 2.0\%$	0.0014	0.012	0.009	0.014	0.014	0.014
$\rho = 2.5\%$	0.0014	0.011	0.008	0.014	0.014	0.014
$\rho = 3.0\%$	0.0014	0.009	0.007	0.013	0.015	0.013
$\rho = 3.5\%$	0.0013	0.007	0.006	0.012	0.014	0.012
$\rho = 4.0\%$	0.0014	0.005	0.005	0.011	0.012	0.011
$\rho = 4.5\%$	0.0012	0.005	0.004	0.010	0.011	0.009
$\rho = 5.0\%$	0.0011	0.004	0.004	0.008	0.010	0.009



(a) Gaussian noise attacks with different STD



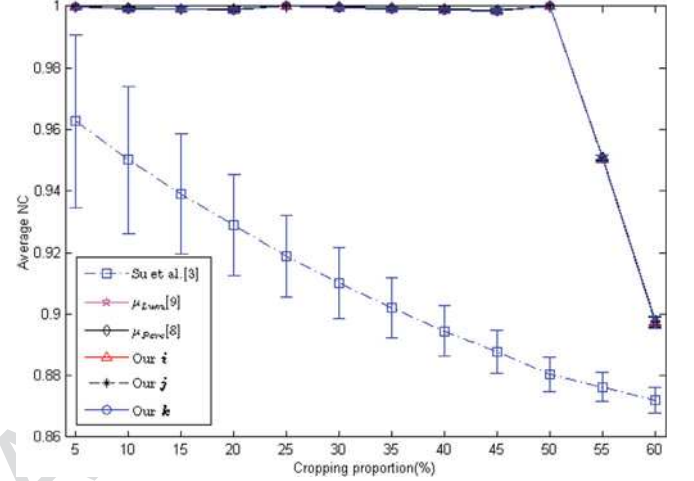
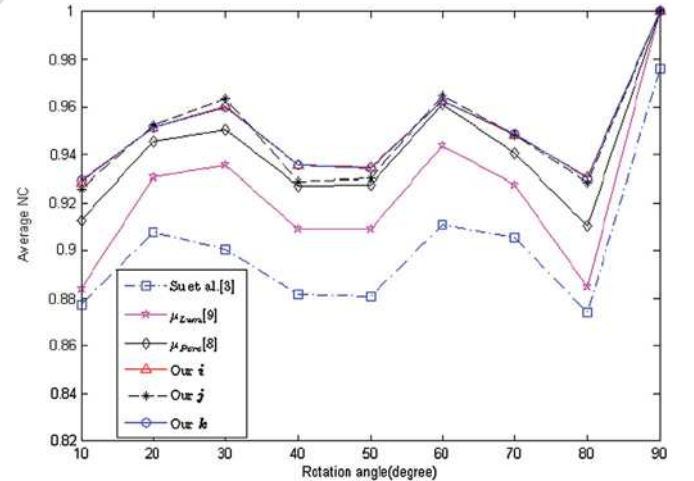
(b) Salt-and-pepper noise attack with different densities

Fig. 11. Average and standard deviation of NC values of color watermarks extracted from the watermarked image test set after noise adding attacks.

robust than the one of Su et al. Notice that for a scaling factor such as $\lambda = 20\%$, extracted watermarks are seriously degraded with NC values not more than 0.8. However, extracted watermarks are provided with high quality when $\lambda \geq 60\%$, especially for $\lambda \geq 100\%$. Besides, all the six tested algorithms including the one of Su et al. are very stable (see Table 5).

5.3.3. Influence of the symmetry constraints

As last experiments, we propose to verify the gain of performance achieved when respecting the symmetry constraints. To do so, we applied quaternion based algorithms similarly as in the tests

**Fig. 12.** Color watermarks NC values in average and accompanied of their standard deviation after a cropping attacks.**Fig. 13.** Average NC values of color watermarks extracted from the image test set attacked after rotation attacks of different angles.

about watermark imperceptibility in Section 5.2. We modulate a set of QDFT coefficients so as to reach a specific PSNR value while respecting or not the symmetry constraints. More clearly, when algorithms do not satisfy symmetry constraints, they modulate independently QDFT coefficients' components, which has the effect to almost double the available capacity and the number of times the watermark is repeated compared to the original version of our algorithm, and to introduce a watermark energy loss. The detection process remains the same for all algorithms and relies on the NC computed between extracted and embedded watermarks. Considering the watermarking of Fig. 4(a) with Fig. 4(k), Fig. 8(a) and

Table 4

NC standard deviation for color watermarks extracted from the image test set after rotation attacks of different angles.

Rotation attack with angle θ	Su et al. [3]	μ_{Lum} [9]	μ_{Perc} [8]	Our i	Our j	Our k
$\theta = 10^\circ$	0.044	0.032	0.035	0.041	0.038	0.041
$\theta = 20^\circ$	0.050	0.044	0.035	0.038	0.036	0.038
$\theta = 30^\circ$	0.052	0.044	0.037	0.036	0.033	0.036
$\theta = 40^\circ$	0.050	0.042	0.040	0.042	0.037	0.042
$\theta = 50^\circ$	0.049	0.042	0.040	0.042	0.037	0.042
$\theta = 60^\circ$	0.052	0.041	0.030	0.034	0.032	0.034
$\theta = 70^\circ$	0.053	0.044	0.035	0.038	0.036	0.038
$\theta = 80^\circ$	0.043	0.032	0.035	0.041	0.038	0.041
$\theta = 90^\circ$	0.033	0	0	0	0	0

Table 5

NC standard deviation for color watermarks extracted from the image test set after different scaling attacks.

Scaling attack with factor λ	Su et al. [3]	μ_{Lum} [9]	μ_{Perc} [8]	Our i	Our j	Our k
$\lambda = 20\%$	0.029	0.005	0.004	0.006	0.009	0.008
$\lambda = 40\%$	0.029	0.010	0.026	0.009	0.031	0.007
$\lambda = 60\%$	0.046	0.043	0.041	0.042	0.040	0.041
$\lambda = 80\%$	0.039	0.037	0.036	0.032	0.031	0.031
$\lambda = 100\%$	0.033	0	0	0	0	0
$\lambda = 120\%$	0.035	0.030	0.027	0.021	0.019	0.020
$\lambda = 140\%$	0.037	0.025	0.022	0.028	0.027	0.027
$\lambda = 160\%$	0.033	0	0	0	0.003	0
$\lambda = 180\%$	0.033	0	0	0.007	0.006	0.006
$\lambda = 200\%$	0.033	0	0	0	0	0

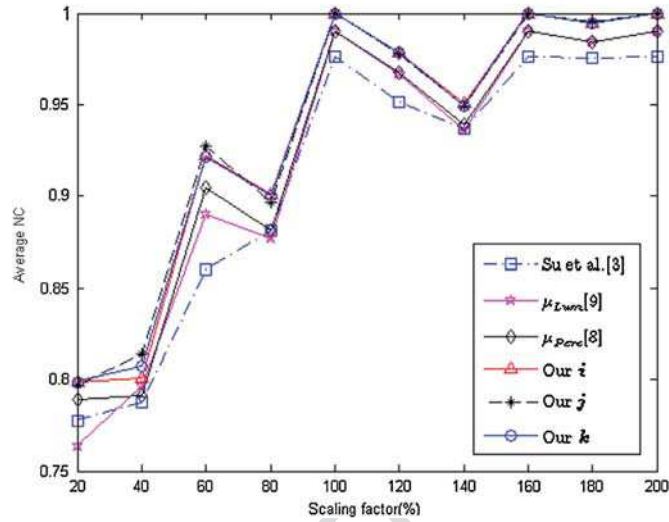
**Fig. 14.** NC values of extracted color watermarks after the scaling attacks. Results are given in average over our test image set.

Fig. 8(b) provide the NC values we obtained for different watermarked image quality under a JPEG attack of quality factor $q = 60$. As it can be seen, watermark repetition does not compensate the watermark energy loss whatever the unit pure quaternion exploited. NC values drop dramatically if embedding does not respect symmetrical constraints. Thus respecting these constraints provides better robustness performance. Notice that if some results of low PSNR are not given, especially for the algorithm using μ_{Lum} . The reason is the same as in the second experiment in Section 5.2: no available QDFT coefficient components can be used to reach such PSNR values.

From all the above experiments, it appears that our algorithms based on i , j or k provide better robustness than other two quaternion algorithms based on μ_{Perc} [8] and μ_{Lum} [9,13,16] while better preserving the image quality at the same time. Among all the algorithms, our based on j performs best. This demonstrates the advantage of taking the four dimensions of the QDFT domain into account.

6. Conclusions

In this paper, by establishing the link between the components of QDFT coefficients with the DFT of color channels, we have shown how to avoid the watermark energy loss during its embedding, and specified the symmetry constraints one must respect when modulating QDFT coefficients in order to fully utilize the 4-D quaternion frequency domain. Considering a general embedding framework, we have demonstrated the overall gain of performance we can achieve for different unit pure quaternions. Compared to previous quaternion based algorithms [8,9,13], we can reach a better trade-off between fidelity and robustness using our algorithms, especially if this one is based on the unit pure quaternion $\mu = j$. As another contribution, we also have improved the method of Bas et al. [8].

Acknowledgments

This work was supported by the National Basic Research Program of China under Grant 2011CB707904, the NSFC under Grants 61271312, 61232016, 61103141, 61173141, 61105007 and 61272421, the Ministry of Education of China under Grant 20110092110023, the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant 13KJB520015, a Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions, and Science Research Foundation of Nanjing University of Information Science & Technology under Grant 20110430.

References

- [1] F. Hartung, M. Kutter, Multimedia watermarking techniques, *Proc. IEEE* 87 (7) (1999) 1079–1107.
- [2] A. Trémeau, D. Muselet, Recent trends in color image watermarking, *J. Imaging Sci. Technol.* 53 (1) (2009) 0102011–01020115.
- [3] Q.T. Su, Y.G. Niu, X.X. Liu, Y. Zhu, Embedding color watermarks in color images based on Schur decomposition, *Opt. Commun.* 285 (7) (2012) 1792–1802.
- [4] S.C. Pei, C.M. Cheng, Color image processing by using binary quaternion-moment preserving thresholding technique, *IEEE Trans. Image Process.* 8 (5) (1999) 22–35.
- [5] T.A. Ell, S.J. Sangwine, Hypercomplex Fourier transforms of color images, *IEEE Trans. Image Process.* 16 (1) (2007) 22–35.

- [6] O.N. Subakan, B.C. Vemuri, A quaternion framework for color image smoothing and segmentation, *Int. J. Comput. Vis.* 91 (3) (2011) 233–250.
- [7] B.J. Chen, H.Z. Shu, H. Zhang, G. Chen, C. Toumoulin, J.L. Dillenseger, L.M. Luo, Quaternion Zernike moments and their invariants for color image analysis and object recognition, *Signal Process.* 92 (2) (2012) 308–318.
- [8] P. Bas, N.L. Nihan, J.M. Chassery, Color image watermarking using quaternion Fourier transform, in: *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, ICASSP 2003*, vol. III, 2003, pp. 521–524.
- [9] S.H. Jiang, J.Q. Zhang, B. Hu, Content based image watermarking algorithm in hypercomplex frequency domain, *Syst. Eng. Electron.* 31 (9) (2009) 1773–1778 (in Chinese).
- [10] T.K. Tsui, X.P. Zhang, D. Androutsos, Quaternion image watermarking using the spatio-chromatic Fourier coefficients, in: *Proc. 14th Annual ACM Int. Conf. Multimedia (MULTIMEDIA '06)*, 2006, pp. 149–152.
- [11] T.K. Tsui, X.P. Zhang, D. Androutsos, Color image watermarking using multi-dimensional Fourier transforms, *IEEE Trans. Inf. Forensics Secur.* 3 (1) (2008) 16–28.
- [12] X.J. Ma, Y. Xu, L. Song, X.K. Yang, H. Burkhardt, Color image watermarking using local quaternion Fourier spectral analysis, in: *Proc. 2008 IEEE Int. Conf. Multimedia and Expo (ICME 2008)*, 2008, pp. 233–236.
- [13] W.J. Wang, W. Zuo, X.M. Yan, New gray-scale watermarking algorithm of color images based on quaternion Fourier transform, in: *Proc. 3th Int. Workshop on Advanced Computational Intelligence*, 2010, pp. 593–596.
- [14] J. Sun, J.Y. Yang, Quaternion frequency watermarking algorithm for color images, in: *Proc. 2010 Int. Conf. Multimedia Technology (ICMT 2010)*, 2010, pp. 1–4.
- [15] J. Sun, J.Y. Yang, A secure color images watermarking algorithm based on holistic quaternion operation, *Adv. Inf. Sci. Serv. Sci.* 3 (10) (2011) 363–374.
- [16] X.Y. Wang, C.P. Wang, H.Y. Yang, P.P. Niu, A robust blind color image watermarking in quaternion Fourier transform domain, *J. Syst. Softw.* 86 (2) (2013) 255–277.
- [17] D. Kundur, D. Hatzinakos, Digital watermarking for telltale tamper proofing and authentication, *Proc. IEEE* 87 (7) (1999) 1167–1180.
- [18] W.R. Hamilton, *Elements of Quaternions*, Longmans Green, London, UK, 1866.
- [19] T.A. Ell, *Hypercomplex spectral transforms*, Ph.D. dissertation, Minnesota University, Minneapolis, 1992.
- [20] T. Bülow, *Hypercomplex spectral signal representations for image processing and analysis*, Ph.D. dissertation, University of Kiel, Kiel, Germany, 1999.
- [21] S.J. Sangwine, Fourier transforms of colour images using quaternion or hypercomplex numbers, *Electron. Lett.* 32 (1) (1996) 1979–1980.
- [22] S.C. Pei, J.J. Ding, J.H. Chang, Efficient implementation of quaternion Fourier transform, convolution, and correlation by 2-D complex FFT, *IEEE Trans. Signal Process.* 49 (11) (2001) 2783–2797.
- [23] C.S. Lu, S.W. Sun, C.Y. Hsu, P.C. Chang, Media hash-dependent image watermarking resilient against both geometric attacks and estimation attacks based on false positive-oriented detection, *IEEE Trans. Multimed.* 8 (4) (2006) 668–685.
- [24] M.R. Soheili, Redundant watermarking using wavelet packets, in: *Proc. 2008 IEEE/ACS Int. Conf. Computer Systems and Applications (AICCSA 2008)*, 2008, pp. 591–598.
- [25] Y.N. Zhang, C.Y. Wen, Y. Zhang, Y.C. Soh, On the choice of consistent canonical form during moment normalization, *Pattern Recognit. Lett.* 24 (16) (2003) 3205–3215.
- [26] H. Zhang, H.Z. Shu, G. Coatrieux, J. Zhu, Q.M.J. Wu, Y. Zhang, H.Q. Zhu, L.M. Luo, Affine Legendre moment invariants for image watermarking robust to geometric distortions, *IEEE Trans. Image Process.* 20 (8) (2011) 2189–2199.